



Embeddings of automata

Marie-Pierre Béal, Sylvain Lombardy, Dominique Perrin

► To cite this version:

Marie-Pierre Béal, Sylvain Lombardy, Dominique Perrin. Embeddings of automata. 2007. hal-00627816

HAL Id: hal-00627816

<https://hal.science/hal-00627816>

Submitted on 29 Sep 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Embeddings of automata

Marie-Pierre Béal, Sylvain Lombardy and Dominique Perrin

January 28, 2008

Abstract

We prove that any local automaton is contained in a complete one. The proof uses Nasu's masking lemma. We use the same technique to give a new proof that any weakly deterministic irreducible automaton is contained in a complete one.

Contents

1	Introduction	1
2	Equivalences of automata	2
3	Local automata	8
4	Weakly deterministic automata	15

1 Introduction

The problem of embedding a finite automaton into one having specified properties is an old one in automata theory (see [10] for example). We will consider here the problem of embedding an automaton into a complete one for several families of automata including local and weakly deterministic automata.

The problem of embedding an unambiguous irreducible automaton into a complete one has been solved in [1] using a construction due to Ehrenfeucht and Rosenberg [4]. This result has a formulation in terms of codes. Indeed, the stabilizer of a state in an unambiguous automaton is a free submonoid. Thus the above embedding implies that any rational code is contained in a maximal one.

A local automaton is by definition such that a bounded information about the past and the future is enough to determine the present state. Formally there are integers λ, ρ such that for any pair of paths $p \xrightarrow{u} q \xrightarrow{v} r$ and $p' \xrightarrow{u} q' \xrightarrow{v} r'$ having the same label with $|u| = \lambda$ and $|v| = \rho$ one has $q = q'$. A subautomaton of a local automaton is obviously still local but conversely it is not clear how

one can add transitions or states to a local automaton until it is complete, in the sense that any word on the underlying alphabet is the label of a path.

The problem of completing local automata has already been considered in [7], where a method is given which allows one to complete a local deterministic automaton when it is possible. We prove here that any local automaton is always contained in a complete one. The stabilizer of a state in a local automaton is the star of a *locally parsable code* (also called codes with *finite synchronization delay*). The previous result gives an alternative proof of a result of Bruyère according to which any locally parsable rational code is included in a maximal one [3], see also [2].

Our proof relies on a result known as Nasu's masking lemma [9]. This result deals with the embedding of a shift of finite type into another one. Given a graph G , the edge shift on G is the set X_G of biinfinite paths in G . The result states that if X_G can be embedded in X_H , then there is a graph K such that X_K is isomorphic to X_H and G is contained in K . We use this result in the particular case where X_H is the full shift but the same proof can be used to obtain a more general result with an arbitrary subshift instead of the full shift.

We also use Nasu's masking lemma to give a new proof of a result of [1] concerning weakly deterministic automata, which are such that the knowledge of the starting state and of a prefix of length d of the label of a path determines its first edge. These automata are also called right-closing in symbolic dynamics. We give a new proof using Nasu's lemma that any d -weakly deterministic automaton is contained in a complete one. This result can be reformulated in terms of codes. The basis of the stabilizer of a state in a weakly deterministic automaton is, by definition, a *weakly prefix code* (also called codes with *finite deciphering delay*).

We have written this paper in a self-contained form and in particular we adapt the proof of Nasu's lemma to our notation.

2 Equivalences of automata

We denote by $\mathcal{A} = (Q, E)$ a finite automaton on the alphabet A with Q as set of states and $E \subset Q \times A \times Q$ as set of edges. We consider all states as both initial and terminal. The automaton is said to be *complete* (with respect to the alphabet A) if for any word $w \in A^*$ there exists a path labelled w . It is said to be *unambiguous* if for any $p, q \in Q$ and w in A^* there is at most one path from p to q labelled w . It is said to be *irreducible* if for any $p, q \in Q$ there is at least one word w such that there is a path from p to q labelled w .

An automaton is said to be *essential* if any state has at least one incoming edge and at least one outgoing edge. Clearly an irreducible automaton is essential.

An automaton $\mathcal{A} = (P, E)$ on the alphabet A is a *subautomaton* of an automaton $\mathcal{B} = (Q, F)$ on the alphabet A , if $P \subset Q$ and $E \subseteq F \cap (P \times A \times P)$.

An automaton is *deterministic* if for each state p and each letter a there is at most one edge labelled a going out of p . We use the notation $p \cdot a = q$ to

express the fact that there is an edge $p \xrightarrow{a} q$ in a deterministic automaton. It can be shown that an irreducible deterministic automaton is complete if and only if for each state p and letter a there is exactly one edge labelled a going out of p .

The *transition matrix* of an automaton $\mathcal{A} = (Q, E)$ is the $Q \times Q$ matrix with elements in the set $\mathfrak{P}(A)$ of subsets of A defined for $p, q \in Q$ by

$$M_{p,q} = \{a \in A \mid \text{there is an edge } (p, a, q) \text{ in } E\}.$$

The elements of M can be considered as elements of the semiring $\mathfrak{P}(A^*)$ of subsets of A^* , where 0 is the empty set and 1 is the set containing the empty word. Such matrices can therefore be multiplied.

Let $\mathcal{A} = (P, E)$ and $\mathcal{B} = (Q, F)$ be two automata on the alphabet A . Let M be the transition matrix of \mathcal{A} and let N be the transition matrix of \mathcal{B} . We say that \mathcal{A} and \mathcal{B} are *elementary equivalent* if there exist a $(P \times Q)$ -matrix R and a $(Q \times P)$ -matrix S both with elements in $\mathfrak{P}(A) \cup \{1\}$, where 1 is the empty word, such that $M = RS$ and $N = SR$. We also say that \mathcal{A} is (R, S) -elementary equivalent to \mathcal{B} .

This notion is classical in symbolic dynamics. It is usually formulated for subshifts of finite type (see [6] p. 225). Our definition is a particular case of the notion of symbolic elementary equivalence for sofic shifts introduced in [8].

For a $P \times Q$ -matrix R with elements in $\mathfrak{P}(A) \cup \{1\}$, we say that a triple $(p, a, q) \in P \times (A \cup \{1\}) \times Q$ is an R -edge when $a \in R_{p,q}$. Thus when \mathcal{A} is (R, S) -equivalent to \mathcal{B} , each edge of \mathcal{A} is the sequence of an R -edge and a consecutive S -edge, and each edge of \mathcal{B} is a sequence of an S -edge and a consecutive R -edge. The R -edges and S -edges can be considered as edges of a bipartite graph called the *auxiliary graph* of the equivalence (it is called a *bipartite code* in [8] and [6, p. 355]).

Example 1 Let \mathcal{A} and \mathcal{B} be the automata represented on Figure 1.

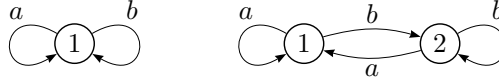


Figure 1: Two elementary equivalent automata \mathcal{A} (on the left) and \mathcal{B} (on the right).

The transition matrices M and N of \mathcal{A} and \mathcal{B} are

$$M = [a + b], \quad N = \begin{bmatrix} a & b \\ a & b \end{bmatrix}.$$

Let

$$R = \begin{bmatrix} a & b \end{bmatrix}, \quad S = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Then $M = RS$ and $N = SR$. Thus \mathcal{A} and \mathcal{B} are elementary equivalent. The auxiliary graph is shown in Figure 2 (with continuous lines).

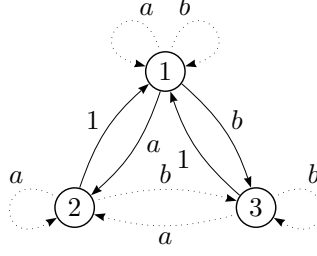


Figure 2: The auxiliary graph (with continuous edges).

Let $\mathcal{A} = (Q, E)$ be an automaton. The *input* of a state $q \in Q$ is the set of pairs (p, a) such that one has $p \xrightarrow{a} q$. Its *output* is the set of pairs (a, r) such that one has $q \xrightarrow{a} r$.

An *input merge equivalence* is an equivalence on the set Q of states of \mathcal{A} such that for any pair p, p' of equivalent states, any letter a and any state q , one has

- (i) $p \xrightarrow{a} q$ if and only if $p' \xrightarrow{a} q$,
- (ii) $q \xrightarrow{a} p, q \xrightarrow{a} p'$ implies $p = p'$.

Thus, in such an equivalence, two equivalent and distinct states have the same output and disjoint inputs.

The quotient of $\mathcal{A} = (Q, E)$ by such an equivalence is the automaton $\overline{\mathcal{A}} = (\overline{Q}, \overline{E})$ with states the set \overline{Q} of equivalence classes, and edges the induced edges on the classes, *i.e.* $(p, a, q) \in E$ for some $p, q \in Q$ implies $(\overline{p}, a, \overline{q})$ is in \overline{E} .

Thus, in the quotient the output of a class of states is the common output of its elements and its input is the union, or merge, of the inputs of its elements (whence the name of an input merge equivalence). We say that $\overline{\mathcal{A}}$ is obtained from \mathcal{A} by an input merge. We also say that \mathcal{A} is obtained from $\overline{\mathcal{A}}$ by an *input split*.

Note that if \mathcal{A} is an unambiguous essential automaton, there is a largest input merge equivalence. It is defined by $q \equiv q'$ if q and q' have the same output. Indeed, since \mathcal{A} is essential, if $q \equiv q'$, they have the same non-empty output. Since it is unambiguous, they have disjoint inputs.

Example 2 Let \mathcal{A}, \mathcal{B} be the automata of Example 1. The automaton \mathcal{A} is obtained from \mathcal{B} by an input merge.

An *output merge equivalence* on the set of states of an automaton \mathcal{A} is defined symmetrically. It is an equivalence on the set Q of states such that equivalent and distinct states have the same input and disjoint outputs. The quotient $\overline{\mathcal{A}}$ is defined in the same way by merging the outputs of the elements of a class. We say that $\overline{\mathcal{A}}$ is obtained from \mathcal{A} by an output merge. We also say that \mathcal{A} is obtained from $\overline{\mathcal{A}}$ by an *output split*.

Observe that the quotient of an automaton \mathcal{A} by an input or an output merge equivalence is unambiguous if and only if \mathcal{A} is unambiguous.

The notions of input and output merge equivalence are classical in symbolic dynamics. They are usually formulated for subshifts of finite type (see [6, p. 225]). The extension of the definitions to sofic shifts is due to Nasu [8]; see also [5].

The following result is an element of William's Classification Theorem (see [6]).

Proposition 1 *If the automaton \mathcal{A} is obtained from the automaton \mathcal{B} by an input (or output) merge, then \mathcal{A} and \mathcal{B} are elementary equivalent.*

Proof We treat the case of an input merge. Let $\mathcal{B} = (Q, E)$ and $\mathcal{A} = (\overline{Q}, \overline{E})$. Let M and N be the transition matrices of \mathcal{A} and \mathcal{B} . Let R be the $\overline{Q} \times Q$ -matrix defined for $p, q \in Q$ by

$$R_{\overline{p}, q} = \{a \in A \mid \text{there is an edge } (p, a, q) \text{ in } E\}$$

(note that R is well-defined because of the definition of an input merge). Let S be the $Q \times \overline{Q}$ -matrix defined by

$$S_{p, \overline{q}} = \begin{cases} 1 & \text{if } \overline{p} = \overline{q} \\ 0 & \text{otherwise} \end{cases}$$

Then $M = RS$ and $N = SR$. Thus \mathcal{A} and \mathcal{B} are elementary equivalent. \square

Proposition 2 *Let \mathcal{A} and \mathcal{B} be two irreducible (R, S) -elementary equivalent automata. Then, either R has elements in $\mathfrak{P}(A)$ and S in $\{0, 1\}$ or conversely.*

Proof Set $\mathcal{A} = (P, E)$ and $\mathcal{B} = (Q, F)$. Suppose that there are $p \in P$, $a \in A$, and $q \in Q$ such that $a \in R_{pq}$. Let $p' \in P$ and $q' \in Q$. We denote $s = S_{q'p'}$. Let us show that, if $s \neq 0$ then $s = 1$. Since \mathcal{B} is irreducible, there is path $q \xrightarrow{u} q'$ labelled by $u = a_1 \dots a_n$. We have $a_i = s_i r_i$, for $1 \leq i \leq n$, with $r_i \in R_{p_i q_i}$, and $s_i \in S_{q_{i-1} p_i}$, and thus the path

$$p \xrightarrow{a} q_0 \xrightarrow{s_1} p_1 \xrightarrow{r_1} q_1 \xrightarrow{s_2} \dots \xrightarrow{s_n} p_n \xrightarrow{r_n} q_n \xrightarrow{s} p',$$

with $q_0 = q$ and $q_n = q'$. Since as_1 is the label of an edge in E , $s_1 = 1$. Since $s_1 r_1$ is the label of an edge in F , $r_1 \in A$ and so on. Finally $r_n \in A$ and thus $s = 1$.

This shows that if one entry of R contains an element of A then all the entries of S are 0 or 1. Furthermore, if all the entries of S are 0 or 1, then all non-null entries of R are in $\mathfrak{P}(A)$. The proof of the symmetrical case is similar. \square

We denote by $L(\mathcal{A})$ the set of words labelling a finite path in \mathcal{A} . Two automata \mathcal{A} and \mathcal{B} are said to be *equivalent* if $L(\mathcal{A}) = L(\mathcal{B})$.

We say that two automata \mathcal{A} and \mathcal{B} are *strongly equivalent* if there is a sequence $\mathcal{A}_0, \mathcal{A}_1, \dots, \mathcal{A}_n$ of automata such that $\mathcal{A}_0 = \mathcal{A}$, $\mathcal{A}_n = \mathcal{B}$ and \mathcal{A}_i is elementary equivalent to \mathcal{A}_{i+1} for $1 \leq i \leq n-1$.

Proposition 3 *Two strongly equivalent essential automata are equivalent.*

Proof Let $\mathcal{A} = (P, E)$ and $\mathcal{B} = (Q, F)$ be (R, S) -elementary equivalent automata on the alphabet A . Let $w = a_1 \dots a_n \in L(\mathcal{A})$. Since \mathcal{A} is essential, there is a path

$$p_{-1} \xrightarrow{a_0} p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots p_{n-1} \xrightarrow{a_n} p_n \xrightarrow{a_{n+1}} p_{n+1},$$

with $a_0, a_{n+1} \in A$. For each $i = 0, \dots, n+1$, we have $a_i = r_i s_i$ with $r_i \in R_{p_{i-1} q_i}$ and $s_i \in S_{q_i p_i}$, and thus a path

$$p_{-1} \xrightarrow{r_0} q_0 \xrightarrow{s_0} p_0 \xrightarrow{r_1} q_1 \xrightarrow{s_1} p_1 \dots p_{n-1} \xrightarrow{r_n} q_n \xrightarrow{s_n} p_n \xrightarrow{r_{n+1}} q_{n+1} \xrightarrow{s_{n+1}} p_{n+1}$$

using alternately entries of R and S . Then either $r_0 = a_0, \dots, r_{n+1} = a_{n+1}$ and $s_0 = \dots = s_{n+1} = 1$ or $r_0 = \dots = r_{n+1} = 1$ and $s_0 = a_0, \dots, s_{n+1} = a_{n+1}$. We consider the first case, the other one being symmetrical. Since SR is the transition matrix of \mathcal{B} , we have in \mathcal{B} a path

$$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \dots q_n \xrightarrow{a_{n+1}} q_{n+1}.$$

This shows that $wa_{n+1} \in L(\mathcal{B})$ and thus $w \in L(\mathcal{B})$. \square

We will now describe a construction due to Nasu (see [6, p. 354], Lemma 10.2.3). It associates to automata $\mathcal{A}_1, \mathcal{A}_2, \mathcal{B}_1$ such that \mathcal{A}_1 is elementary equivalent to \mathcal{A}_2 and \mathcal{A}_1 is a subautomaton of \mathcal{B}_1 , an automaton \mathcal{B}_2 elementary equivalent to \mathcal{B}_1 and such that \mathcal{A}_2 is a subautomaton of \mathcal{B}_2 . The automaton \mathcal{B}_2 is called the *Nasu embedding* of \mathcal{A}_2 with respect to $\mathcal{A}_1, \mathcal{B}_1$.

Set $\mathcal{A}_1 = (P_1, E_1)$, $\mathcal{A}_2 = (P_2, E_2)$ and $\mathcal{B}_1 = (Q_1, F_1)$. Let R, S be matrices such that RS and SR are respectively the transition matrices of \mathcal{A}_1 and \mathcal{A}_2 . Let $Q_2 = P_2 \cup (F_1 \setminus E_1)$. We define a $Q_1 \times Q_2$ -matrix \widehat{R} and a $Q_2 \times Q_1$ -matrix \widehat{S} as follows.

Suppose first that R has 0, 1 entries and S entries in $\mathfrak{P}(A)$. Then, for $q_1 \in Q_1$ and $q_2 \in Q_2$, we define

$$\widehat{R}_{q_1, q_2} = \begin{cases} R_{q_1, q_2} & \text{if } q_1 \in P_1, q_2 \in P_2 \\ 1 & \text{if } q_2 \in F_1 \setminus E_1 \text{ and } o(q_2) = q_1 \\ 0 & \text{otherwise} \end{cases}$$

and

$$\widehat{S}_{q_2, q_1} = S_{q_2, q_1} \cup \{l(q_2) \mid q_2 \in F_1 \setminus E_1, e(q_2) = q_1\},$$

where $o(q_2)$ denotes the origin of q_2 , $l(q_2)$ its label, and $e(q_2)$ its end.

An illustration of this construction is given in Example 3.

In the case where R has entries in $\mathfrak{P}(A)$ and S has 0, 1 entries, we define

$$\widehat{R}_{q_1, q_2} = R_{q_1, q_2} \cup \{a \in A \mid q_2 \in F_1 \setminus E_1, o(q_2) = q_1, l(q_2) = a\}$$

and

$$\widehat{S}_{q_2, q_1} = \begin{cases} S_{q_2, q_1} & \text{if } q_1 \in P_1, q_2 \in P_2 \\ 1 & \text{if } q_2 \in F_1 \setminus E_1 \text{ and } e(q_2) = q_1 \\ 0 & \text{otherwise} \end{cases}.$$

It is clear that in both cases, $\widehat{R}\widehat{S}$ is the transition matrix of \mathcal{B}_1 . The automaton $\mathcal{B}_2 = (Q_2, F_2)$ is defined by its transition matrix which is $\widehat{S}\widehat{R}$. To sum up, we have the following statement.

Proposition 4 *Let $\mathcal{A}_1, \mathcal{A}_2, \mathcal{B}_1$ be automata such that \mathcal{A}_1 is a subautomaton of \mathcal{B}_1 and \mathcal{A}_1 is elementary equivalent to \mathcal{A}_2 . Let \mathcal{B}_2 be the Nasu embedding of \mathcal{A}_2 with respect to $\mathcal{A}_1, \mathcal{B}_1$. Then \mathcal{A}_2 is a subautomaton of \mathcal{B}_2 and \mathcal{B}_1 is elementary equivalent to \mathcal{B}_2 . Moreover, if \mathcal{A}_2 and \mathcal{B}_1 are irreducible (resp. essential), then \mathcal{B}_2 is irreducible (resp. essential).*

Example 3 Figure 3 represents the Nasu embedding \mathcal{B}_2 of \mathcal{A}_2 with respect to $\mathcal{A}_1, \mathcal{B}_1$. The automata \mathcal{A}_1 and \mathcal{A}_2 are the elementary equivalent automata of Example 1. If A_i (resp. B_i) denotes the transition matrix of the automaton \mathcal{A}_i

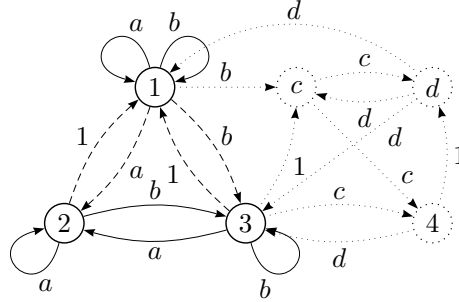


Figure 3: The Nasu embedding \mathcal{B}_2 of \mathcal{A}_2 with respect to $\mathcal{A}_1, \mathcal{B}_1$. The automata $\mathcal{A}_1 = (P_1, E_1)$ with $P_1 = \{2, 3\}$ and $\mathcal{A}_2 = (P_2, E_2)$ with $P_2 = \{1\}$ are represented with continuous edges. We have $\mathcal{B}_1 = (Q_1, F_1)$ with $Q_1 = \{2, 3, 4\}$ and $\mathcal{B}_2 = (Q_2, F_2)$ with $Q_2 = \{1, c, d\}$. The automaton \mathcal{B}_2 has two additional states c, d corresponding to the additional edges $(3, c, 4)$ and $(4, d, 3)$ of \mathcal{B}_1 . The transitions defined by R and S are represented with dashed edges. Additional states and edges are represented with dots.

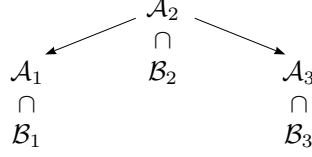
(resp. \mathcal{B}_i) of Figure 3, we have

$$A_1 = \begin{bmatrix} a & b \\ a & b \end{bmatrix} = RS = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} a & b \end{bmatrix}, \quad A_2 = [a + b] = SR.$$

$$B_1 = \begin{bmatrix} a & b & 0 \\ a & b & c \\ 0 & d & 0 \end{bmatrix} = \widehat{R}\widehat{S} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a & b & 0 \\ 0 & 0 & c \\ 0 & d & 0 \end{bmatrix},$$

$$B_2 = \begin{bmatrix} a + b & b & 0 \\ 0 & 0 & c \\ d & d & 0 \end{bmatrix} = \widehat{S}\widehat{R} = \begin{bmatrix} a & b & 0 \\ 0 & 0 & c \\ 0 & d & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The following result will be used in Section 4.



Proposition 5 *Let $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{B}_3$ be essential automata such that \mathcal{A}_1 is an input merge of \mathcal{A}_2 , \mathcal{A}_3 is an output merge of \mathcal{A}_2 and \mathcal{A}_3 is a subautomaton of \mathcal{B}_3 .*

Let \mathcal{B}_2 be the Nasu extension of \mathcal{A}_2 with respect to $\mathcal{A}_3, \mathcal{B}_3$. Then \mathcal{A}_1 is a subautomaton of the maximal input merge \mathcal{B}_1 of \mathcal{B}_2 .

Proof Set $\mathcal{A}_i = (P_i, E_i)$ for $i = 1, 2, 3$ and let $\mathcal{B}_i = (Q_i, F_i)$ for $i = 2, 3$. Let M_i be the transition matrix of \mathcal{A}_i for $i = 1, 2, 3$. Set $M_2 = SR$, $M_3 = RS$ and let \hat{R}, \hat{S} be the matrices corresponding to the Nasu extension of \mathcal{A}_2 . We have $Q_2 = P_2 \cup (F_3 \setminus E_3)$ and for $q_2 \in Q_2$, $q_3 \in Q_3$,

$$\hat{S}_{q_2, q_3} = S_{q_2, q_3} \cup \{l(q_2) \mid q_2 \in F_3 \setminus E_3, e(q_2) = q_3\}$$

and

$$\hat{R}_{q_3, q_2} = \begin{cases} R_{q_2, q_3} & \text{if } q_2 \in P_2, q_3 \in P_3 \\ 1 & \text{if } q_2 \in F_3 \setminus E_3 \text{ and } o(q_2) = q_3 \\ 0 & \text{otherwise} \end{cases}$$

Let $p_2, p'_2 \in P_2$ be two states of \mathcal{A}_2 that have the same output. Let us show that they also have the same output in \mathcal{B}_2 . For this, suppose that (p_2, a, r) is an edge of \mathcal{B}_2 with $r \in F_3 \setminus E_3$. Since the transition matrix of \mathcal{B}_2 is $\hat{S}\hat{R}$, there is a state $q \in Q_3$ such that $a \in \hat{S}_{p_2 q}$ and $\hat{R}_{qr} = 1$. It follows from the definition of \hat{S} that q is in P_3 (indeed if $q \in Q_3 \setminus P_3$, then $p_2 \in F_3 \setminus E_3$, a contradiction). Since \mathcal{A}_3 is obtained from \mathcal{A}_2 by an output merge, there is a $p \in P_2$ such that $R_{q, p} = 1$ and thus an edge (p_2, a, q) in E_2 . Since p_2, p'_2 have the same input in \mathcal{A}_2 , we have also an edge (p'_2, a, q) in E_2 . Thus there is an edge (p'_2, a, r) in F_2 , which proves the assertion.

This shows that the automaton \mathcal{B}_1 obtained from \mathcal{B}_2 by merging the states with the same output, has \mathcal{A}_1 as a subautomaton. \square

3 Local automata

An automaton is said to be (λ, ρ) -local if for all pairs of paths $p \xrightarrow{u} q \xrightarrow{v} r$ and $p' \xrightarrow{u} q' \xrightarrow{v} r'$ with $|u| = \lambda$ and $|v| = \rho$, one has $q = q'$. The automaton is said to be local if it is (λ, ρ) -local for some $\lambda, \rho \geq 0$. An automaton which has a single state is $(0, 0)$ -local.

Thus in a local automaton, one can recover a biinfinite path from its label using a sliding window of fixed size.

As a particular case of a local automaton, an *edge automaton* is such that all edges have distinct labels. An edge automaton is both $(0, 1)$ and $(1, 0)$ -local.

A word w is said to be a *constant* for an automaton \mathcal{A} if for $p, p', q, q' \in P$ such that $p \xrightarrow{w} q$ and $p' \xrightarrow{w} q'$ one has also $p \xrightarrow{w} q'$ and $p' \xrightarrow{w} q$. The empty word is a constant only when the automaton has a single state.

Observe that when the automaton is deterministic, a constant is a word w such that the set $\text{Im}(w) = \{q \in P \mid p \xrightarrow{w} q \text{ for some } p \in P\}$ has at most one element.

Note that if \mathcal{A} is an irreducible, complete, deterministic local automaton with more than one state, then

$$\bigcap_{a \in A} \text{Im}(a) = \emptyset, \quad (1)$$

where $\text{Im}(a) = \{p \cdot a \mid p \in P\}$ (this property appears in [7]). Indeed, suppose that $q \in \bigcap_{a \in A} \text{Im}(a)$. Let w be a word of maximal length which is not a constant. Then for each letter $a \in A$, the word aw is a constant and thus $\text{Im}(aw)$ has only one element. Thus for any state p we have $p \cdot w = q \cdot w$. Indeed, since \mathcal{A} is essential, there is a letter a such that $p \in \text{Im}(a)$. Then $p, q \in \text{Im}(a)$, which implies $p \cdot w = q \cdot w$. This contradicts the fact that w is not a constant.

Example 4 The automaton of Figure 4 is $(1, 1)$ -local. All nonempty words but a are constants.

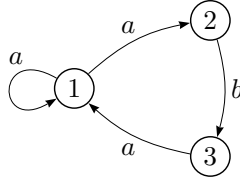


Figure 4: A $(1, 1)$ -local automaton.

Note that if \mathcal{A} is a (λ, ρ) -local automaton, then an input split of \mathcal{A} is a $(\lambda + 1, \rho)$ -local automaton, and an output split of \mathcal{A} is a $(\lambda, \rho + 1)$ -local automaton.

Proposition 6 *An irreducible automaton is local if and only if it is unambiguous and there exists an integer n such that any word of length n is a constant.*

Proof Suppose first that \mathcal{A} is (λ, ρ) -local. Let c, c' be two paths from p to q labelled w . Since \mathcal{A} is irreducible, there exist states r, s and paths $r \xrightarrow{u} p, q \xrightarrow{v} s$ with $|u| = \lambda$ and $|v| = \rho$. Since \mathcal{A} is (λ, ρ) -local, the paths c, c' coincide. Thus \mathcal{A} is unambiguous. Suppose next that $p \xrightarrow{w} q$ and $p' \xrightarrow{w} q'$ with $|w| = \lambda + \rho$. Let $w = uv$ with $|u| = \lambda$ and $|v| = \rho$. Let r, r' be such that $p \xrightarrow{u} r \xrightarrow{v} q$ and $p' \xrightarrow{u} r' \xrightarrow{v} q'$. Since \mathcal{A} is (λ, ρ) -local, we have $r = r'$. This implies that we also have paths $p \xrightarrow{w} q'$ and $p' \xrightarrow{w} q$. Thus the condition of the statement is true with $n = \lambda + \rho$.

Conversely, let us assume that any word of length n is a constant and that \mathcal{A} is unambiguous. Let us show that \mathcal{A} is (n, n) -local. Suppose that $p \xrightarrow{u} q \xrightarrow{v} r$ and $p' \xrightarrow{u} q' \xrightarrow{v} r'$ with $|u| = |v| = n$. Then, since u and v are constants, we also have $p \xrightarrow{u} q' \xrightarrow{v} r$. Since \mathcal{A} is unambiguous, this forces $q = q'$, whence the conclusion. \square

The least integer n such that any word of length n is a constant is called the *order* of the automaton. Note that a (λ, ρ) -local automaton has order at most $\lambda + \rho$. It can however be strictly less than $\lambda + \rho$. An unambiguous automaton with order n is (n, n) -local.

Let $\lambda, \rho \geq 0$. The *free* (λ, ρ) -local automaton is the automaton with set of states consisting of pairs (u, v) of words, with u of length λ and v of length ρ , and edges the triples $((u, v), a, (u', v'))$ such that there are letters b, c with $uvc = bu'v'$ and a is the first letter of v when v is nonempty. It is clear that this automaton is (λ, ρ) -local.

The free $(n, 0)$ -local automaton is usually known as the *de Bruijn automaton* of order n .

Lemma 1 *Let n be a nonnegative integer such that $\lambda, \rho \leq n$, the free (λ, ρ) -local automaton has order n .*

Proof Let $\mathcal{F}_{(\lambda, \rho)}$ be the free (λ, ρ) -local automaton. Let $(u, v) \xrightarrow{w} (x, y)$ and $(u', v') \xrightarrow{w} (x', y')$ be two paths labelled w of length n in $\mathcal{F}_{(\lambda, \rho)}$, with u, u', x, x' of length λ and v, v', y, y' of length ρ . Since $\lambda, \rho \leq n$, v, v' are prefixes of w and x, x' are suffixes of w . Hence $v = v'$ and $x = x'$ and there are paths $(u, v) \xrightarrow{w} (x, y')$ and $(u', v) \xrightarrow{w} (x, y)$ in $\mathcal{F}_{(\lambda, \rho)}$. \square

Example 5 The free $(1, 1)$ -local automaton on the alphabet $\{a, b\}$ is represented on Figure 5. The label of an edge going out of a state is its second letter.

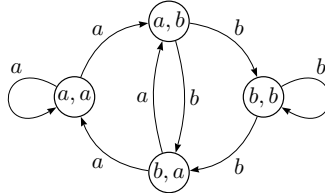


Figure 5: The free $(1, 1)$ -local automaton. It has order 1.

The following result shows that any (λ, ρ) -local automaton is strongly equivalent to a subautomaton of the free (λ, ρ) -local automaton.

Proposition 7 *If \mathcal{A} is (λ, ρ) -local automaton, there is a sequence $\mathcal{A}_0, \mathcal{A}_1, \dots, \mathcal{A}_{\lambda+\rho}$ of (λ, ρ) -local automata such that*

- (i) $\mathcal{A}_0 = \mathcal{A}$.
- (ii) For $i = 0, \dots, \lambda - 1$, \mathcal{A}_i is obtained from \mathcal{A}_{i+1} by an input merge.
- (iii) For $i = \lambda, \dots, \lambda + \rho - 1$, \mathcal{A}_i is obtained from \mathcal{A}_{i+1} by an output merge.
- (iv) $\mathcal{A}_{\lambda+\rho}$ is a subautomaton of the free (λ, ρ) -local automaton.

Proof Let $\mathcal{A} = (P, E)$. We define for $1 \leq i \leq \lambda$, $\mathcal{A}_i = (P_i, E_i)$ where P_i is the set of pairs (u, q) with $u \in A^i$ and $q \in Q$ such that there is a path labelled u leading to q in \mathcal{A} . There is an edge labelled a from (u, p) to (v, q) in \mathcal{A}_i if $u = bu'$, $v = u'a$ for $b \in A$ and $(p, a, q) \in E$. For $j = 1, \dots, \rho$, let $\mathcal{A}_{\lambda+j} = (P_{\lambda+j}, E_{\lambda+j})$ where $P_{\lambda+j}$ is the set of triples $(u, p, v) \in A^\lambda \times P \times A^j$ such that there is a path labelled u leading to p and a path labelled v leaving p in \mathcal{A} . There is an edge labelled a from (u, p, v) to (w, q, t) if and only if $u = bu'$, $w = u'a$, $v = av'$, $t = v'c$ and $(p, a, q) \in E$.

For $0 \leq i \leq \lambda - 1$, the equivalence θ_{i+1} on P_{i+1} defined by $(u, p) \equiv (u', p')$ if $p = p'$ and u, u' differ at most by the first letter is an input merge. Similarly, for $\lambda \leq i \leq \lambda + \rho - 1$, the equivalence θ_{i+1} on P_{i+1} defined by $(u, p, v) \equiv (u', p', v')$ if $u = u'$, $p = p'$ and v, v' differ at most by their last letter, is an output merge. This shows that conditions (ii) and (iii) are satisfied.

Finally, since \mathcal{A} is (λ, ρ) -local, in a state (u, p, v) of $\mathcal{A}_{\lambda+\rho}$, the state p is determined by (u, v) . Thus condition (iv) is also satisfied.

□

Observe that if \mathcal{A} is moreover supposed to be irreducible, then all the automata \mathcal{A}_i constructed as above are also irreducible.

We will prove the following result.

Theorem 2 *Any local unambiguous (local irreducible) automaton is a subautomaton of a complete (local irreducible) one with the same order.*

The proof uses Nasu embeddings.

Proposition 8 *Let \mathcal{A}_1 and \mathcal{A}_2 be automata such that \mathcal{A}_2 is obtained from \mathcal{A}_1 by input (or output) merge. If \mathcal{A}_1 is a subautomaton of a local automaton \mathcal{B}_1 , then the Nasu embedding of \mathcal{A}_2 with respect to \mathcal{A}_1 and \mathcal{B}_1 , is a local automaton with the same order as \mathcal{B}_1 .*

Proof We treat the case of an input merge. The case of an output merge is symmetrical. Let $\mathcal{B}_1 = (Q_1, F_1)$ be a local automaton containing $\mathcal{A}_1 = (P_1, E_1)$ as a subautomaton. Assume that $\mathcal{A}_2 = (P_2, E_2)$ is obtained from \mathcal{A}_1 by input merge. Let $F = F_1 \setminus E_1$, $Q_2 = P_2 \cup F$ and let $\mathcal{B}_2 = (Q_2, F_2)$ be the Nasu embedding of \mathcal{A}_2 with respect to $\mathcal{A}_1, \mathcal{B}_1$.

Let R, S be the matrices defined as in the proof of Proposition 1. The transition matrix of \mathcal{A}_1 is equal to RS while the transition matrix of \mathcal{A}_2 is equal to SR . We have $a \in S_{\bar{p}, q}$ if $(p, a, q) \in E_1$ and $R_{p, \bar{p}} = 1$. The additional \hat{S} -edges are the triples (f, a, q) such that $f = (p, a, q)$ is in F . The additional

\widehat{R} -edges are the triples $(p, 1, f)$ such that f begins with p . An example of this construction is described in Figure 3.

Let π be the map defined on the set \mathcal{P} of paths of \mathcal{B}_1 of length at least 2 onto the set of nonempty paths of \mathcal{B}_2 as follows. If $(e_0, e_1, \dots, e_n) \in \mathcal{P}$ with $e_i = (q_i, a_i, q_{i+1})$, we define $\pi(e_0, e_1, \dots, e_n) = (g_0, g_1, \dots, g_{n-1})$ with $g_i = (p_i, a_i, p_{i+1})$ defined by $p_i = \bar{q}_i$ if $e_i \in E_1$ and $p_i = e_i$ if $e_i \in F$.

We claim that π is a surjective map from the set \mathcal{P} onto the set of nonempty paths of \mathcal{B}_2 . We have to verify that each g_i is an edge of \mathcal{B}_2 . We distinguish four cases.

- (i) If $p_i, p_{i+1} \in P_2$. Then g_i is an edge of \mathcal{A}_2 and thus of \mathcal{B}_2 . It is the concatenation of an S -edge and an R -edge.
- (ii) If $p_i \in P_2$ and $p_{i+1} \in F$. Then $p_i = \bar{q}_i$ and $p_{i+1} = e_{i+1}$. Thus g_i is the concatenation of an S -edge and an $(\widehat{R} - R)$ -edge.
- (iii) If p_i, p_{i+1} are in F , then $p_i = e_i$ and $p_{i+1} = e_{i+1}$. Thus $g_i = (p_i, a_i, p_{i+1})$ is the concatenation of an $(\widehat{S} - S)$ -edge and an $(\widehat{R} - R)$ -edge.
- (iv) If $p_i \in F$ and $p_{i+1} \in P_2$ then $p_i = e_i$. Thus g_i is the concatenation of an $(\widehat{S} - S)$ -edge and an R -edge.

Thus π is well defined. Moreover, one can verify that for any nonempty path c in \mathcal{B}_2 there exists $c' \in \mathcal{P}$ such that $\pi(c') = c$. Thus π is surjective.

Let n be the order of \mathcal{B}_1 . Let $c = p \xrightarrow{w} q$ and $c' = p' \xrightarrow{w} q'$ be two paths in \mathcal{B}_2 with $|w| = n$. Then there exist paths $d = r \xrightarrow{w} s \xrightarrow{a} t$ and $d' = r' \xrightarrow{w} s' \xrightarrow{a'} t'$ in \mathcal{B}_1 , with $a, a' \in A$, such that $\pi(d) = c$ and $\pi(d') = c'$. Since \mathcal{B}_1 has order n , w is a constant for \mathcal{B}_1 . Thus we have also paths $e = r \xrightarrow{w} s' \xrightarrow{a'} t'$ and $e' = r' \xrightarrow{w} s \xrightarrow{a} t$ in \mathcal{B}_1 . It is easy to verify that $\pi(e)$ is a path from p to q' and $\pi(e')$ a path from p' to q . Thus w is a constant for \mathcal{B}_2 . \square

Proof of Theorem 2. Let \mathcal{A} be a local unambiguous automaton of order n . Without loss of generality, one can assume that \mathcal{A} is (λ, ρ) -local with $\lambda, \rho \leq n$ (for instance one can choose $\lambda = \rho = n$). By Proposition 7 there is a sequence $\mathcal{A}_0, \mathcal{A}_1, \dots, \mathcal{A}_{\lambda+\rho}$ of automata such that $\mathcal{A} = \mathcal{A}_0$, $\mathcal{A}_{\lambda+\rho}$ is a subautomaton of the free (λ, ρ) -local automaton and each \mathcal{A}_i is a merge of \mathcal{A}_{i+1} . By Lemma 1, the free (λ, ρ) -local automaton has order n . Since $\mathcal{A}_{\lambda+\rho}$ is included in the complete (λ, ρ) -local automaton, we may build using repeatedly Proposition 8 a sequence $\mathcal{B}_{\lambda+\rho}, \dots, \mathcal{B}_1, \mathcal{B}_0$ of complete unambiguous local automata with order n such that \mathcal{A}_i is contained in \mathcal{B}_i for $i = \lambda + \rho, \dots, 0$. Thus $\mathcal{B} = \mathcal{B}_0$ is a complete local automaton with order n containing \mathcal{A} . It is clear that if \mathcal{A} is irreducible, then \mathcal{B} is irreducible. \square

Example 6 Let $\mathcal{A} = \mathcal{A}_0$ be the deterministic automaton represented on the left of Figure 6. It is $(3, 0)$ -local and has order 3. Indeed, the action of a, b on the set of states represented on the right of Figure 6 shows that any word of length 3 has an image of cardinality at most 1.

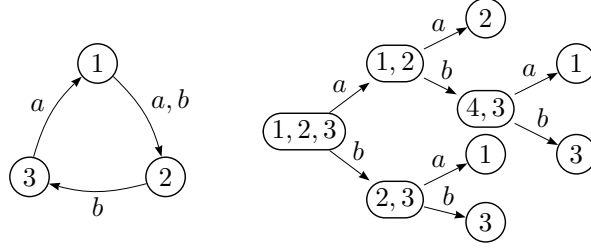


Figure 6: A local automaton \mathcal{A} and the action on subsets.

Note that \mathcal{A} cannot be completed in a local deterministic automaton. Indeed, we have $2 \in \text{Im}(a) \cap \text{Im}(b)$ contrary to (1). We have represented on Figure 7 a split \mathcal{A}_1 of \mathcal{A} , obtained by an input split of state 2 in two states 5 and 7. This automaton can easily be completed as indicated on Figure 7.

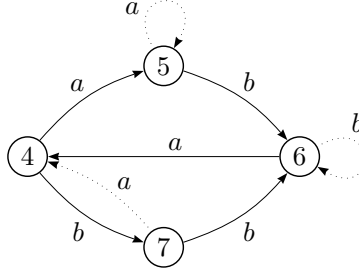


Figure 7: A split \mathcal{A}_1 of \mathcal{A} and its completion \mathcal{B}_1 with the dotted edges.

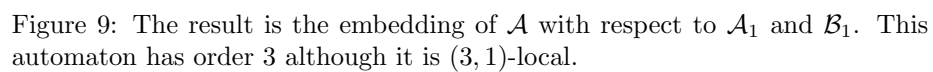
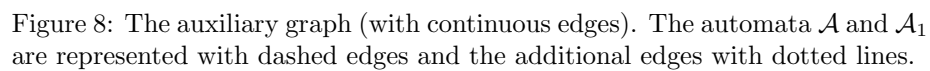
The matrices R, S such that the transition matrix of \mathcal{A}_1 is RS and the transition matrix of \mathcal{A} is SR are

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad S = \begin{bmatrix} 0 & a & 0 & b \\ 0 & 0 & b & 0 \\ a & 0 & 0 & 0 \end{bmatrix}.$$

The auxiliary graph is shown on Figure 8. The final result is shown on Figure 9. It is a complete local automaton containing \mathcal{A} .

To check that it is complete, one can apply the subset construction to build an equivalent deterministic automaton. Since \mathcal{A} is irreducible and local, this construction may be started from any state. Then we can check that each state has outgoing edges carrying each letter.

Theorem 2 can be generalized as follows. Let \mathcal{A} be a local automaton. Any local automaton \mathcal{B} such that $L(\mathcal{B}) \subseteq L(\mathcal{A})$ is a subautomaton of a local automaton \mathcal{C} strongly equivalent to \mathcal{A} . Moreover, if \mathcal{A} is an automaton of order



at most one, then the order of \mathcal{C} can be chosen equal to the order of \mathcal{B} . Theorem 2 corresponds to the case where the automaton \mathcal{A} has just one state.

Example 7 Let \mathcal{A} be the $(1,0)$ -local automaton represented on the right of Figure 10. The automaton \mathcal{B} represented on the left with continuous lines is $(1,1)$ -local. One has $L(\mathcal{B}) \subset L(\mathcal{A})$. Adding the dotted edge labelled a to \mathcal{B} gives a $(1,1)$ -local automaton \mathcal{C} which is equivalent to the automaton \mathcal{A} .

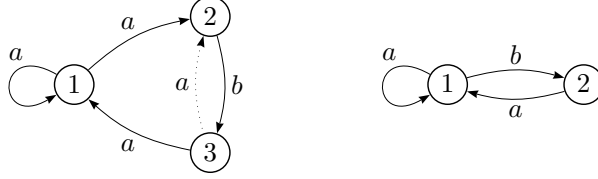


Figure 10: The local automata \mathcal{B}, \mathcal{C} on the left (with the additional dotted edge for \mathcal{C}), and \mathcal{A} on the right.

4 Weakly deterministic automata

An automaton \mathcal{A} is said to have *delay* d if for any pair of paths

$$p \xrightarrow{a} q \xrightarrow{z} r, \quad p \xrightarrow{a} q' \xrightarrow{z} r',$$

with $a \in A$, $z \in A^*$ and $|z| = d$ then $q = q'$. We say that the automaton is *weakly deterministic* if it has delay d for some integer $d \geq 0$. The *minimal delay* of an automaton \mathcal{A} is the least integer d such that \mathcal{A} has delay d .

Weakly deterministic automata are also known in symbolic dynamics as *right-closing* automata.

Thus a deterministic automaton has delay 0. Observe that if \mathcal{A} has delay d , then for any word w , and for any pair of paths

$$p \xrightarrow{w} q \xrightarrow{z} r, \quad p \xrightarrow{w} q' \xrightarrow{z} r',$$

with $w, z \in A^*$ and $|z| = d$, the paths $p \xrightarrow{w} q$ and $p \xrightarrow{w} q'$ are equal.

Proposition 9 *Let \mathcal{A} be an essential automaton. If \mathcal{A} is local, it is weakly deterministic. If it is weakly deterministic, it is unambiguous.*

Proof Suppose first that \mathcal{A} is (λ, ρ) -local. Let $p \xrightarrow{a} q \xrightarrow{z} r$ and $p \xrightarrow{a} q' \xrightarrow{z} r'$ be paths with $|z| = \rho$. Since \mathcal{A} is essential, there exists a path $s \xrightarrow{y} p$ with $|ya| \geq \lambda$. Since we have paths $s \xrightarrow{ya} q \xrightarrow{z} r$ and $s \xrightarrow{ya} q' \xrightarrow{z} r'$, we have $q = q'$. Thus \mathcal{A} has delay ρ .

Suppose next that \mathcal{A} has delay d . Two paths from p to q with the same label may be extended to the right arbitrarily. This forces them to coincide. Thus \mathcal{A} is unambiguous. \square

We will use the following result.

Proposition 10 *Let \mathcal{A}_1 and \mathcal{A}_2 be (R, S) -strongly equivalent irreducible automata. If the automata \mathcal{A}_1 is weakly deterministic, then \mathcal{A}_2 also, and their minimal delays differ at most by one.*

Proof Let us suppose that \mathcal{A}_2 has delay $d - 1$ and show that \mathcal{A}_1 has delay d . We treat the case where the entries of R are in $\mathfrak{P}(A)$ and the entries of S are in $\{0, 1\}$. The other case is similar.

Let $p_0 \xrightarrow{a_0} p_1 \xrightarrow{a_1} \dots \xrightarrow{a_d} p_{d+1}$ and $p_0 \xrightarrow{a_0} p'_1 \xrightarrow{a_1} \dots \xrightarrow{a_d} p'_{d+1}$ be two paths in \mathcal{A}_1 of length $d + 1$ with the same origin and the same label $a_0 a_1 \dots a_d$, with $a_i \in A$. Since \mathcal{A}_1 is irreducible, there exists an S -edge $q_0 \rightarrow p_0$. Thus we have, in the auxiliary graph, paths $q_0 \rightarrow p_0 \xrightarrow{a_0} q_1 \rightarrow p_1 \xrightarrow{a_1} q_2 \dots \xrightarrow{a_d} p_{d+1}$ and $q_0 \rightarrow p_0 \xrightarrow{a_0} q'_1 \rightarrow p'_1 \xrightarrow{a_1} q'_2 \dots \xrightarrow{a_d} p'_{d+1}$. Since \mathcal{A}_2 has delay $(d - 1)$, we have $q_1 = q'_1$ and $q_2 = q'_2$. This implies $p_1 = p'_1$. Thus \mathcal{A}_1 has delay d . \square

The following statement shows that an input merge cannot increase the delay.

Proposition 11 *Let \mathcal{A}_1 be an input merge of \mathcal{A}_2 . Then \mathcal{A}_1 and \mathcal{A}_2 have the same minimal delay.*

Proof The delay of a (input or output) merge cannot be larger than the original one. Hence if \mathcal{A}_2 has delay d , then \mathcal{A}_1 has delay d . Conversely, suppose that \mathcal{A}_1 has delay d . Let $p \xrightarrow{a} q \xrightarrow{u} r$ and $p \xrightarrow{a} q' \xrightarrow{u} r'$ be two paths in \mathcal{A}_2 where $a \in A$ and $u \in A^d$. We then have paths $\bar{p} \xrightarrow{a} \bar{q} \xrightarrow{u} \bar{r}$ and $\bar{p} \xrightarrow{a} \bar{q}' \xrightarrow{u} \bar{r}'$ in \mathcal{A}_1 . Since \mathcal{A}_1 has delay d , we have $\bar{q} = \bar{q}'$. Since the inputs of q and q' are not disjoint, we have $q = q'$. Thus \mathcal{A}_2 has delay d . \square

The following result, which already appears in [1], is the analogue of Theorem 2 for weakly deterministic automata. The proof uses again Nasu embeddings.

Theorem 3 *A weakly deterministic irreducible automaton is a subautomaton of a complete one with the same delay.*

Observe that, contrary to Theorem 2, the above statement is trivial if the result is not required to be irreducible.

We first prove the following result which shows that a weakly deterministic irreducible automaton is strongly equivalent to a deterministic irreducible automaton.

Proposition 12 *If \mathcal{A} is a weakly deterministic irreducible automaton with delay $d \geq 1$, there exist automata \mathcal{A}_1 and \mathcal{A}_2 such that \mathcal{A} is an input merge of \mathcal{A}_1 , \mathcal{A}_2 is an output merge of \mathcal{A}_1 and \mathcal{A}_2 has delay $d - 1$.*

Proof Set $\mathcal{A}_0 = \mathcal{A}$ and $\mathcal{A}_i = (P_i, E_i)$ for $i = 0, 1, 2$. Let $P_1 = E_0$ and let E_1 be the set of triples (e, a, f) with $e, f \in E_0$ and $a \in A$ such that e, f are consecutive edges of \mathcal{A} and a is the label of f . In this way, there is an input merge from \mathcal{A}_1 to \mathcal{A}_0 obtained by merging the states with the same third component.

Let \mathcal{A}_2 be the automaton obtained from \mathcal{A}_1 by merging the states having the same input. By Proposition 11 and since \mathcal{A}_0 has delay d , \mathcal{A}_1 has delay d . Let us verify that \mathcal{A}_2 has delay $d - 1$. Consider two paths $\alpha \xrightarrow{a} \beta \xrightarrow{u} \gamma$ and $\alpha \xrightarrow{a} \beta' \xrightarrow{u} \gamma'$ in \mathcal{A}_2 with $a \in A$ and $u \in A^{d-1}$. These paths are the image under the output merge of two paths in \mathcal{A}_1 of the form $(i, b, p) \xrightarrow{a} (p, a, q) \xrightarrow{u} (r, c, s)$ and $(i', b, p') \xrightarrow{a} (p', a, q') \xrightarrow{u} (r', c, s')$. Since (i, b, p) and (i', b, p') have the same input in \mathcal{A}_1 , we have $i = i'$. Thus we obtain the two paths $i \xrightarrow{b} p \xrightarrow{au} s$ and $i \xrightarrow{b} p' \xrightarrow{au} s$ in \mathcal{A} . Since \mathcal{A} has delay d , this forces $p = p'$. But since (p, a, q) and (p, a, q') have the same input in \mathcal{A}_1 , they belong to the same class of the output merge equivalence giving \mathcal{A}_2 . This forces $\beta = \beta'$. \square

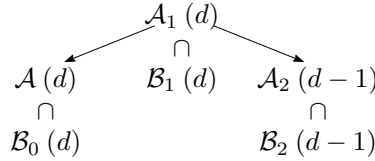


Figure 11: The construction in the proof of Proposition 12. The delay of each automaton is given inside the parenthesis.

Proof of Theorem 3. We use an induction on the minimal delay d of \mathcal{A} . If $d = 0$, then \mathcal{A} is deterministic and the property is clearly true. Suppose $d \geq 1$. Let $\mathcal{A}_0 = \mathcal{A}$ and $\mathcal{A}_1, \mathcal{A}_2$ be the automata given by Proposition 12. The construction is illustrated in Figure 11. By induction, \mathcal{A}_2 is a subautomaton of an irreducible automaton \mathcal{B}_2 with delay $d-1$. Let \mathcal{B}_1 be the Nasu embedding of \mathcal{A}_1 with respect to $(\mathcal{A}_2, \mathcal{B}_2)$. By Proposition 10, \mathcal{A}_1 has delay d . By Proposition 5, there is an input merge \mathcal{B}_0 of \mathcal{B}_1 such that \mathcal{A}_0 is a subautomaton of \mathcal{B}_0 . By Proposition 11, \mathcal{B}_0 has delay d , which concludes the proof. \square

Example 8 Figure 12 illustrates the construction of the completion of a weakly deterministic automaton.

The particular solution given in [1] is not the same as the one described above. Let us explain how it works on this example.

The embedding is based on the observation that an automaton with delay d is complete if and only if for each state p the following condition is satisfied: for each word u of length d such that there is a path labelled u starting at p , there also a path labelled ua starting at p for each $a \in A$.

Since state 2 can be followed by ba but not by bb , we add an edge labelled b from 2 to a new state called b and such that there is an edge labelled b going to state 1. Further, since state b can be followed by ba but not by bb , we add a loop labelled b on state b . The above condition is now satisfied for any state. Let \mathcal{B}_0 be the complete automaton thus obtained.

This automaton is obtained by the previous construction starting with the completion of the deterministic automaton \mathcal{A}_2 shown on the right and working

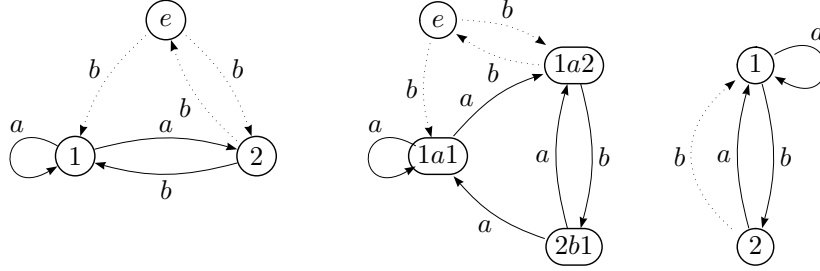


Figure 12: Let $\mathcal{A} = \mathcal{A}_0$ be the automaton with delay 1 represented on the left with continuous lines. The automaton \mathcal{A}_1 is represented at the center. The automaton \mathcal{A}_2 obtained by merging states $1a1$ and $1a2$ is shown on the right. It is deterministic and can be completed by adding an edge e labelled b from state 2 to state 1 represented with dashes (another solution would be to add a loop labelled b on 2). Working backwards, the Nasu embeddings of \mathcal{A}_1 and \mathcal{A}_0 appear with dashes. The result on the left is a complete automaton \mathcal{B}_0 with delay 1 containing \mathcal{A}_0 as a subautomaton.

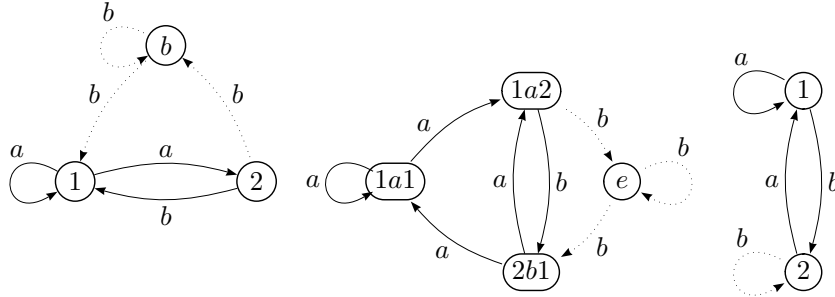


Figure 13: Another embedding of the previous automaton.

backwards as before.

The full statement of the result of [1] is more general and Theorem 3 and its proof can be extended as follows. Let \mathcal{A} be an irreducible edge automaton. Any weakly deterministic irreducible automaton \mathcal{B} such that $L(\mathcal{B}) \subseteq L(\mathcal{A})$ is a subautomaton of a weakly deterministic irreducible automaton \mathcal{C} which is equivalent to \mathcal{A} and has the same delay.

We have not been able to found a way to prove by Nasu extensions the analogous result of [1] concerning bi-closing automata, *i.e.* which have finite delay in both directions.

Acknowledgements: We would like to thank Brian Marcus who pointed out to us the possibility of using Nasu's masking lemma in this problem.

References

- [1] Jonathan Ashley, Brian Marcus, Dominique Perrin, and Selim Tuncel. Surjective extensions of sliding block codes. *SIAM J. Discrete Math.*, 6:582–611, 1993.
- [2] Jean Berstel, Dominique Perrin, and Christophe Reutenauer. *Codes and Automata*. Cambridge University Press, 2007. To appear, see <http://www-igm.univ-mlv.fr/~berstel/LivreCodes/newlivreCodes.pdf>.
- [3] Véronique Bruyère. On maximal codes with bounded synchronization delay. *Theor. Comput. Sci.*, 204:11–28, 1988.
- [4] Andrei Ehrenfeucht and Gregorz Rozenberg. Each regular code is included in a regular maximal code. *RAIRO Informat. Theor.*, 20:89–96, 1983.
- [5] Wolfgang Krieger. On sofic systems. I. *Israel J. Math.*, 48(4):305–330, 1984.
- [6] Douglas A. Lind and Brian H. Marcus. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, 1995.
- [7] Rosalba Montalbano. Local automata and completion. In *STACS 93*, volume 665 of *Lecture Notes in Comput. Sci.*, pages 333–342. Springer verlag, 1993.
- [8] Masakazu Nasu. Topological conjugacy for sofic systems. *Ergodic Theory Dynam. Systems*, 6(2):265–280, 1986.
- [9] Masakazu Nasu. Topological conjugacy for sofic systems and extensions of automorphisms of finite subsystems of topological markov shifts. In *Proceedings of Maryland special year in Dynamics 1986–87*, volume 1342 of *Lecture Notes in Mathematics*, pages 564–607. Springer verlag, 1988.
- [10] Marcel-Paul Schützenberger. A remark on incompletely specified automata. *Information and Control*, 8:373–376, 1965.

Index

- automata
 - elementary equivalent , 3
 - equivalent, 5
 - strongly equivalent, 5
- automaton
 - bi-closing, 18
 - complete, 2
 - delay, 14
 - essential, 2
 - irreducible, 2
 - local, 8
 - minimal delay, 14
 - order, 9
 - right-closing, 15
 - unambiguous, 2
 - weakly deterministic, 14
- auxiliary graph, 3
- bipartite code, 3
- code
 - locally parsable, 2
 - weakly prefix, 2
- constant, 9
- de Bruijn automaton, 10
- deciphering delay, 2
- delay, 14
- edge automaton, 9
- elementary equivalent automata, 3
- equivalent automata, 5
- free local automaton, 10
- input, 3
- input merge equivalence, 4
- input split, 4
- locally parsable, 2
- minimal delay, 14
- Nasu embedding, 6
- order, 9
- output, 3
- output merge equivalence, 4
- output split, 4
- subautomaton, 2
- synchronization delay, 2
- weakly deterministic automaton, 14
- weakly prefix, 2